5. Глобальный еврейский Online Центр [Электронный ресурс]. URL: http://www.jewish.ru/news/israel/2013/01/news994314013.php (дата обращения: 3.10.13)

6. *Безмалый В.* Новое время – новые угрозы // Windows IT Pro/RE: проф. изд., посв. вопросам работы с продуктами семейства Windows и технологиям компании Microsoft. 2012 [Электронный ресурс]. URL: http://www.osp.ru/win2000/2012/08/13033288/ (дата обращения: 03.10.13).

7. *Ледовской В.* Android под прицелом. Беззубая свобода // Anti-Malware.ru – первый в России независимый информационно-аналитический центр, полностью посвященный информационной безопасности. 2012 [Электронный ресурс]. URL: http://www.anti-malware.ru/analytics/Android_under_sight/ (дата обращения: 03.10.13).

УЯЗВИМОСТЬ И ЗАЩИТА АЛГОРИТМА ДИФФИ – ХЕЛЛМАНА

Е. О. Федюшина, М. А. Балашов
(Екатеринбург, УрГУПС, eleno4ka45@mail.ru)

Безопасность систем шифрования зависит от конфиденциальности ключа, используемого в алгоритме шифрования, а не от хранения в тайне самого алгоритма. Многие алгоритмы шифрования общедоступны и были хорошо проверены благодаря этому. Цель данной работы – изучить алгоритм обмена ключей Диффи – Хеллмана как пример шифрования с открытым ключом, выявить его уязвимость и найти способ ее устранения.

Алгоритм Диффи – Хеллмана – алгоритм, позволяющий двум сторонам получить общий секретный ключ, используя незащищенный от прослушивания, но защищенный от подмены канал связи. Этот ключ может быть использован для шифрования дальнейшего обмена с помощью алгоритма симметричного шифрования.

Система Диффи – Хеллмана разрабатывалась для решения проблемы распространения ключей при использовании систем шифрования с секретными ключами. Идея заключалась в том, чтобы применять безопасный метод согласования секретного ключа без передачи ключа каким-либо другим способом. Следовательно, необходи-

мо было найти безопасный способ получения секретного ключа с помощью того же метода связи, для которого разрабатывалась защита.

Определим круг возможностей алгоритма. Предположим, что двум абонентам необходимо провести конфиденциальную переписку, а в их распоряжении нет первоначально оговоренного секретного ключа. Однако между ними существует канал, защищенный от модификации, т. е. данные, передаваемые по нему, могут быть прослушаны, но не изменены (такие условия имеют место довольно часто). В этом случае две стороны могут создать одинаковый секретный ключ, ни разу не передав его по сети, по следующему алгоритму.

Предположим, что обоим абонентам известны некоторые два числа g и p (при этом p – простое число, а g – первообразный корень числа p). Они, впрочем, известны и всем остальным заинтересованным лицам. Например, они могут быть просто фиксированно «зашиты» в программное обеспечение. Для того чтобы создать неизвестный более никому секретный ключ, оба абонента генерируют случайные или псевдослучайные простые числа: первый абонент – число a, второй абонент – число b. Затем первый абонент вычисляет значение $x = g^a(mod\ p)$ и пересылает его второму, а второй вычисляет $y = g^b(mod\ p)$ и передает первому. Злоумышленник получает оба этих значения, но модифицировать их (вмешаться в процесс передачи) не может. На втором этапе первый абонент на основе имеющегося у него a и полученного по сети y вычисляет значение $k = (g^b(mod\ p))^a(mod\ p))$, а второй абонент на основе имеющегося у него b и полученного по сети x вычисляет значение $k = (g^a(mod\ p))^b(mod\ p)$. На самом деле операция возведения в степень переносима через операцию взятия модуля по простому числу (коммутативна в конечном поле), т. е. у обоих абонентов получилось одно и то же число: $k = g^{ab}(mod\ p)$. Его они и могут использовать в качестве секретного ключа, поскольку здесь злоумышленник встретится с проблемой дискретного логарифмирования при попытке выяснить по перехваченным x и y сами числа a и b – это очень и очень ресурсоемкая операция, если числа g, n, a, b выбраны достаточно большими.

Необходимо еще раз отметить, что алгоритм Диффи – Хеллмана работает только на линиях связи, надежно защищенных от мо-

дификации. Если бы он был применим на любых открытых каналах, то давно снял бы проблему распространения ключей и, возможно, заменил собой всю асимметричную криптографию.

Следует заметить, что данный алгоритм уязвим для атак типа man-in-the-middle. Если противник может осуществить активную атаку, т. е. имеет возможность не только перехватывать сообщения, но и заменять их другими, он может перехватить открытые ключи участников x и y, создать свою пару открытого и закрытого ключа и послать каждому из участников свой открытый ключ. После этого каждый участник вычислит ключ, который будет общим с противником, а не с другим участником. Если нет контроля целостности, то участники не смогут обнаружить подобную подмену. Осуществление такой атаки требует большого объема ресурсов, и в реальном мире такие атаки происходят редко.

Рассмотрим на примере алгоритм обмена ключей Диффи – Хеллмана и еще одну его уязвимость.

В качестве пользователей, которым нужно обменяться ключами, выступают Алиса и Боб. Ева – криптоаналитик. Она читает пересылку Алисы и Боба, но не изменяет содержимого их сообщений.

Вначале Алиса и Боб публично определяют p и g. Например, $p = 997$ и $g = 7$ (7 – первообразный корень 997). Потом Алиса выбирает приватное число $a = 221$ и вычисляет $x = 7^{221} (mod\ 997) = 652$, и посылает этот результат публично Бобу. Боб выбирает свое приватное число $b = 313$ и вычисляет $y = 7^{313} (mod\ 997) = 661$, и посылает результат публично Алисе. Алиса берет публичный результат Боба и возводит его в степень своего приватного числа, получает $k = 661^{221} (mod\ 997) = 7^{313 \times 221} (mod\ 997) = 46$ – общий секретный ключ. Боб берет публичный результат Алисы и возводит его в степень своего приватного числа $k = 652^{313} (mod\ 997) = 7^{221 \times 312} (mod\ 997) =$ $= 46$ – тот же самый общий ключ.

Итак, приватной является информация $a = 221$, $b = 313$ и $k = 46$. Публичная информация, которую может перехватить Ева: $p = 997$, $g = 7$, $x = 652$ и $y = 661$. Задача Евы узнать общий секретный ключ k. Если Ева определит a или b, она беспрепятственно сможет найти k. Определим, например, a.

Для начала рассмотрим задачу дискретного логарифмирования $a^b \equiv c(mod\ n)$, так как алгоритм Диффи – Хеллмана основан именно на нем. Так как во всех необходимых формулах присутствует mod – остаток от деления, необходима формула, раскрывающая его, т. е. обратная mod. Выведем эту формулу. Например, возьмем $n = 17$ и $a = 3$ (3 – первообразный корень 17). И найдем степень 3, при которой остаток от деления на 17 будет равен, например, 13, т. е. $3^b \equiv 13(mod\ 17)$. Зная смысл данного сравнения, сделаем вывод, что число 3^b делится на 17 так, что получается некоторое количество целых частей (обозначим их s) и остаток $c = 13$. Тогда получаем новую формулу: $3^b = 17 \times s + 13$. Для общего случая формула примет вид: $a^b = n \times s + c$. Отсюда b можно найти при помощи обычного логарифмирования: $b = \log_3(17 \times s + 13)$. И теперь остается найти такое s, при котором b будет целым числом:

s	b
1	3,095
2	3,505
3	3,786
4	4
5	4,173

Получаем $b = 4$ при $s = 4$. Проверим результат, составив таблицу всех степеней 3 и найдя остатки от деления на 17:

b	c	b	c
1	3	9	14
2	9	10	8
3	10	11	7
4	13	12	4
5	5	13	12
6	15	14	2
7	11	15	6
8	16	16	1

Действительно, при $b = 4$ остаток $c = 13$, т. е. $3^4 \equiv 13 (mod\ 17)$.

Что касается переменной s. От чего зависит ее длина? Так как s – это число целых частей при делении g^a на p (или g^b на p), то чем ближе a и b к p, то тем меньше получится s.

Вернемся к задаче Диффи – Хеллмана и применим выведенные формулы в нашем случае. Заменим a на g и b на a, тем самым a^b на g^a, c на x, n на p. Получим $g^a = p \times s + x$. Отсюда $s = \frac{g^a - x}{p}$. И выразим по уже известной формуле искомое a: $a = \log_g(p \times s + x)$.

Все необходимые формулы у нас есть, теперь можно вернуться к задаче нахождения a по известным p, g, x и y. Для этого составляем таблицу степеней g от 1 до n при $n < p - 1$. Для каждого g^a из этой таблицы находим s по формуле $s = \frac{g^a - x}{p}$. Ответом окажется та степень a, при которой s – целое. В нашем случае s = = 586099950684229201345256120579:: 364198169865551062770052620404291021600302359519520224669077237:: 486807350277769530967800344763226725491218205917157093115968811:: 8695784654659729976311369721 5.

Подставим найденное s в формулу $a = \log_7(997 \times s + 652)$ и получим $a = 221$.

Алисой в качестве a было выбрано число 221, с помощью данного алгоритма мы получили это же число. Зная a и y, Ева может вычислить k. Аналогично можно вычислить b и найти k по b и x.

Распишем алгоритм нахождения a пошагово:

1. Составление таблицы степеней g от 1 до n при $n < p - 1$;

2. Нахождение s по формуле $s = \frac{g^a - x}{p}$;

3. Нахождение a по формуле $a = \log_g(p \times s + x)$.

Какой бы ни была длина используемого ключа, количество операций останется равным 3, за исключением того, что присутствует перебор по степеням. Но это не составит большой проблемы, если расчеты ведутся при помощи программы с заданным ал-

горитмом. Скорость выполнения довольно простых операций достаточно велика, поэтому искомый ключ в итоге все равно будет получен.

Второй вариант: данная работа по взлому алгоритма Диффи – Хеллмана в дальнейшем может быть продолжена, что все-таки приведет к упрощению алгоритма и более быстрому нахождению секретного ключа, так как начало уже положено и уязвимость в алгоритме найдена. Тогда возникает необходимость обезопасить алгоритм от подобного рода взломов.

Так как сложность во взломе алгоритма обусловливается наличием дискретного логарифма, то предположим, что при двойном дискретном логарифмировании сложность взлома алгоритма повысится.

Так же, как и в самом алгоритме Диффи – Хеллмана, выбираем 2 числа: g и p, с тем же условием, что g – первообразный корень простого числа p. Абоненты выбирают числа a и b, вычисляют $x_1 = g^a (mod\ p)$ и $y_1 = g^b (mod\ p)$ соответственно. Далее, не передавая друг другу полученные значения и не сообщая их никому, производят еще одно аналогичное вычисление, но уже вместо g используют значения x_1 и y_1 соответственно: $x_2 = x_1^a (mod\ p)$ и $y_2 = y_1^b (mod\ p)$. И только на этом этапе обмениваются полученными значениями x_2 и y_2. Каждый из абонентов дважды повторяет процедуру дискретного логарифмирования с полученными значениями, т. е. те же действия, что и в самом начале, но вместо g берутся значения x_2 и y_2: $k_1 = y_2^a (mod\ p)$ и $r_1 = x_2^b (mod\ p)$. После второго вычисления: $k_2 = k_1^a (mod\ p)$ и $r_2 = r_1^b\ (mod\ p)$. В итоге получаем два одинаковых ключа k_2 и r_2.

Рассмотрим этот алгоритм на примере.

Пусть $p = 739$, первообразная от него $g = 3$; $a = 211$, $b = 317$.

Тогда $x_1 = 3^{221} (mod\ 739) = 35$ и $y_1 = 3^{317} (mod\ 739) = 406$.

$x_2 = 35^{221} (mod\ 739) = 643$ и $y_2 = 406^{317} (mod\ 739) = 282$.

Далее абоненты обмениваются значениями x_2 и y_2. И снова повторяют те же действия:

$k_1 = 282^{221} (mod\ 739) = 580$ и $r_1 = 643^{317} (mod\ 739) = 578$.

$k_2 = 580^{221} (mod\ p) = 71$ и $r_2 = 578^{317} (mod\ 739) = 71$.

В итоге получили общий секретный ключ $k_2 = r_2 = 71$.

Надежность алгоритма увеличивается за счет увеличения числа неизвестных в вычислениях при перехвате злоумышленником x_2 и y_2. Например, при перехвате x в первом случае в формуле $x = g^a (mod\ p)$ неизвестным является одно значение a при известных x, g и p. А в случае перехвата x_2 в формуле $x_2 = x_1^a (mod\ p)$ неизвестными будут x_1 и a. И при необходимости нахождения формулы для раскрытия искомого значения a злоумышленник столкнется с проблемой нахождения логарифма в логарифме при условии, что ему не известны промежуточные результаты вычислений, которые могли бы облегчить задачу. Тем самым вычислить секретный ключ будет невозможно.

Чтобы еще повысить криптостойкость алгоритма шифрования, процедуру взятия дискретного логарифма можно повторять и более двух раз, тем самым увеличивая число неизвестных в конечной формуле.